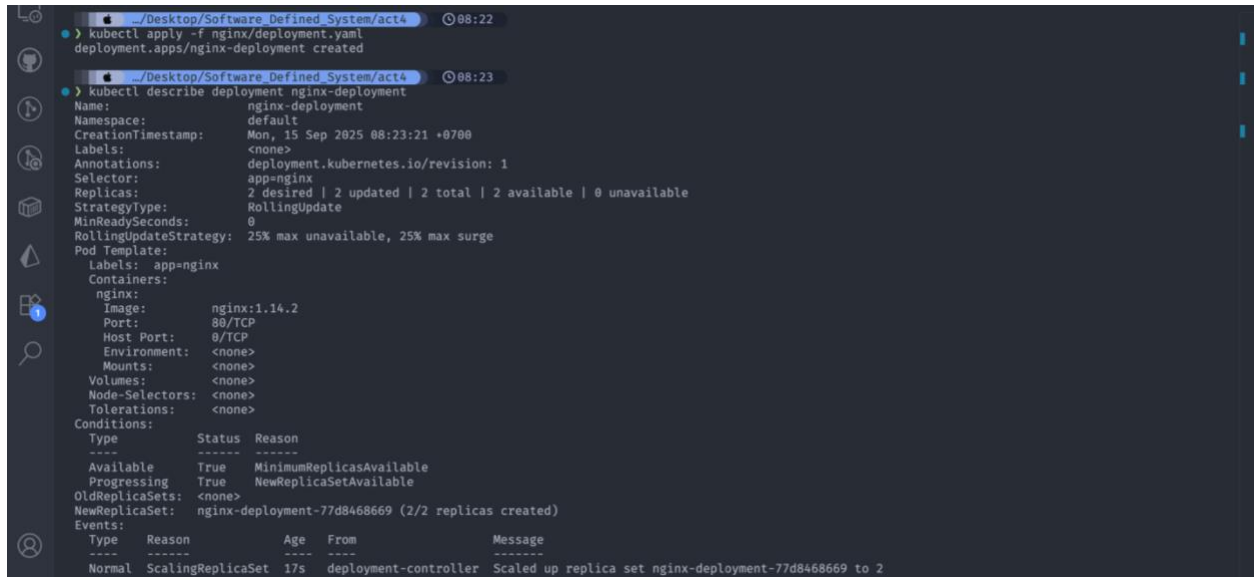


Activity 4

3. Test your cluster with nginx as followed:

- a. Deploy an nginx on your cluster using the deployment file from <https://k8s.io/examples/application/deployment.yaml>
- b. Use kubectl to describe the deployment



```
~/Desktop/Software Defined System/act4 08:22
> kubectl apply -f nginx/deployment.yaml
deployment.apps/nginx-deployment created

~/Desktop/Software Defined System/act4 08:23
> kubectl describe deployment nginx-deployment
Name: nginx-deployment
Namespace: default
CreationTimestamp: Mon, 15 Sep 2025 08:23:21 +0700
Labels: <none>
Annotations: deployment.kubernetes.io/revision: 1
Selector: app=nginx
Replicas: 2 desired | 2 updated | 2 total | 2 available | 0 unavailable
StrategyType: RollingUpdate
MinReadySeconds: 0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
  Labels: app=nginx
  Containers:
    nginx:
      Image: nginx:1.14.2
      Port: 80/TCP
      Host Port: 0/TCP
      Environment: <none>
      Mounts: <none>
      Volumes: <none>
      Node-Selectors: <none>
      Tolerations: <none>
  Conditions:
    Type           Status  Reason
    ----           -
    Available       True    MinimumReplicasAvailable
    Progressing     True    NewReplicaSetAvailable
    OldReplicaSets: <none>
    NewReplicaSet:  nginx-deployment-77d8468669 (2/2 replicas created)
  Events:
    Type      Reason      Age   From               Message
    ----      -
    Normal    ScalingReplicaSet   17s   deployment-controller  Scaled up replica set nginx-deployment-77d8468669 to 2
```

a. & b.

4. Deploy a todo service with redis database as followed:

b. Create a deployment file of a pod with service of the following containers:

- I. todo (release-2.1)
- II. redis

```

c. apiVersion: apps/v1
d. kind: Deployment
e. metadata:
f.   name: todo-service-redis-deployment
g.   labels:
h.     app: todo-service-redis
i. spec:
j.   replicas: 1
k.   selector:
l.     matchLabels:
m.       app: todo-service-redis
n.   template:
o.     metadata:
p.       labels:
q.         app: todo-service-redis
r.     spec:
s.       containers:
t.         - name: todo-service
u.           image: natawut/todo-service:release-2.1
v.           ports:
w.             - containerPort: 8000
x.           env:
y.             - name: REDIS_HOST
z.               value: localhost
aa.
bb.         - name: redis
cc.           image: redis:7-alpine
dd.

```

deployment.yaml

```

~/Desktop/Software_Defined_System/act4 08:27
➤ kubectl apply -f todo-service-redis/deployment.yaml
deployment.apps/todo-service-redis-deployment created

~/Desktop/Software_Defined_System/act4 08:28
➤ kubectl expose deployment todo-service-redis-deployment --type=ClusterIP --port=80 --target-port=8000
service/todo-service-redis-deployment exposed

~/Desktop/Software_Defined_System/act4 08:28
➤ kubectl apply -f todo-service-redis/ingress.yaml
ingress.networking.k8s.io/todo-service-redis-ingress created

```

b.

```

~/Desktop/Software_Defined_System/act4 08:28
> kubectl get deployments
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment    2/2     2             2           6m54s
todo-service-redis-deployment 1/1     1             1           2m14s

~/Desktop/Software_Defined_System/act4 08:30
> kubectl get services
NAME                TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes          ClusterIP   10.96.0.1    <none>        443/TCP   62m
todo-service-redis-deployment ClusterIP   10.102.230.221 <none>        80/TCP   2m17s

~/Desktop/Software_Defined_System/act4 08:30
> kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
nginx-deployment-77d8468669-gb72s   1/1     Running   0           7m3s
nginx-deployment-77d8468669-qbt7p   1/1     Running   0           7m3s
todo-service-redis-deployment-576ff46fd-6r74h 2/2     Running   0           2m23s

```

b. get commands results.

d. Create a deployment file of a pod with service of the following containers:

```

~/Desktop/Software Defined System/act4 08:46
> helm repo add nginx-stable https://helm.nginx.com/stable
"nginx-stable" has been added to your repositories

~/Desktop/Software Defined System/act4 08:46
> helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "nginx-stable" chart repository
Update Complete. *Happy Helming!*

~/Desktop/Software Defined System/act4 08:46
> helm install nginx-ingress nginx-stable/nginx-ingress --set rbac.create=true
NAME: nginx-ingress
LAST DEPLOYED: Mon Sep 15 08:46:23 2025
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
NGINX Ingress Controller 5.1.1 has been installed.

For release notes for this version please see: https://docs.nginx.com/nginx-ingress-controller/releases/
Installation and upgrade instructions: https://docs.nginx.com/nginx-ingress-controller/installation/installing-nic/installation-with-helm/

~/Desktop/Software Defined System/act4 08:46
> kubectl get pods --all-namespaces -l app.kubernetes.io/instance=nginx-ingress
NAMESPACE NAME READY STATUS RESTARTS AGE
default nginx-ingress-controller-5c6b5547fb-fp9xg 1/1 Running 0 14s

~/Desktop/Software Defined System/act4 08:46
> kubectl get services -l app.kubernetes.io/instance=nginx-ingress
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
nginx-ingress-controller LoadBalancer 10.102.177.54 localhost 80:30142/TCP,443:30618/TCP 18s

```

nginx-ingress

```

apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: todo-service-redis-ingress
  namespace: default
  annotations:
    nginx.ingress.kubernetes.io/rewrite-target: /
spec:
  ingressClassName: nginx
  rules:
    - host: localhost
      http:
        paths:
          - path: /
            pathType: Prefix
            backend:
              service:
                name: todo-service-redis-deployment
                port:
                  number: 80

```

ingress.yaml

```

> kubectl apply -f todo-service-redis/ingress.yaml
ingress.networking.k8s.io/todo-service-redis-ingress created

```

apply ingress

```

~/Desktop/Software Defined System/act4 08:30
> kubectl get ingress
NAME                                CLASS  HOSTS  ADDRESS  PORTS  AGE
todo-service-redis-ingress          nginx  localhost  10.1.0.22  80     2m29s

~/Desktop/Software Defined System/act4 08:30
> kubectl describe ingress todo-service-redis-ingress
Name:          todo-service-redis-ingress
Labels:        <none>
Namespace:     default
Address:
Ingress Class: nginx
Default backend: <default>
Rules:
  Host        Path  Backends
  ----        -
  localhost    /    todo-service-redis-deployment:80 (10.1.0.22:8000)
Annotations:  nginx.ingress.kubernetes.io/rewrite-target: /
Events:       <none>

```

describe ingress

e. Test the newly deployment

i. Create a new todo record with

```
{
  "title": "todo-1",
  "detail": "the first todo",
  "duedate": "2022-10-24 11:00:00",
  "tags": [],
  "completed": false
}
```

ii. Get todo list via localhost:80

```

~/Desktop/Software Defined System/act4 08:49
> curl -X POST -d '{"title":"todo-1","detail":"the first todo","duedate":"2024-10-13 11:00:00","tags":[],"completed":false}' http://localhost/
{"id":"0"}INFO: 10.1.0.23:57058 - "POST / HTTP/1.1" 201 Created

~/Desktop/Software Defined System/act4 08:49
> curl localhost
[{"id":"0","title":"todo-1","detail":"the first todo","completed":false,"duedate":"2024-10-13T11:00:00","tags":[]}]INFO: 10.1.0.23:38570 - "GET / HTTP/1.1" 200 OK

```